

# Алгоритм решения задачи

## Описание проблемы:

Дан массив строк. Наша задача состоит в том, чтобы найти общий префикс слов этого массива.

## Предисловие:

Для решения этого задания мы будем использовать встроенные функции `min()` и `max()`. Эти функции применяются к разным типам данных, в том числе и к строкам. Когда речь идет о массивах чисел, интуитивно понятно как они работают. Абсолютно тот же механизм применяется и к строкам: подобный тому, как мы ищем из списка случайных слов первое по алфавиту слово. Находится слово, чья первая (или вторая, может быть третья, если все слова начинаются с одинаковых символов) буква имеет наименьший номер символа в ASCII. К примеру, номер английского символа 'a' находится под номером 97, а 'b' под номером 98, поэтому минимальным словом будет то, которое начинается с символа 'a'. Можно также присвоить параметру `key = length`, тогда минимальное слово – это строка, с минимальным количеством символом.

## Решения задачи:

Для начала мы проверим, является ли входной список пустым. Если да – возвращаем пустую строку:

```
➤ if len(strs) == 0:  
    return ""
```

Дальше найдем “минимальное” и “максимальное” слово:

```
➤ min_st = min(strs)  
    max_st = max(strs)
```

Почему мы ищем эти слова? Для будущего их сравнения. Нам интересны первые символы строк, такой способ позволяет сравнивать только две строки из всех имеющихся в списке. А что на счет других?

```
➤ for i in range(len(min_st)):  
    if max_st[i] != min_st[i]:  
        return max_st[:i]  
    return min_st
```

И так, если первый символ `min_st` и `max_st` не совпадают мы возвращаем пустую строку. Нам не нужно проверять символы других строк, потому что у нас уже имеется несовпадающая пара.

Если же первый символ этих слов совпадает, сравниваем другой, если и эти совпадают – третий и так далее. Как только находим несовпадающие символы, выходим из цикла и возвращаем имеющийся общий префикс строк.

Также возможен случай, когда одно слово является частью второго, например: ['west', 'western']. В таком случаи переменная `max_st` = 'western', а `min_st` = 'west' потому что `western` длинее чем `west`.\*\* Именно поэтому количество итераций в нашем цикле равно длине минимального слова.\*\* В таком случаи наша функция вернет слово `покороче`.